On the Influence of Selection Schemes on the Genetic Diversity in Genetic Algorithms*

Michael Affenzeller, Stephan Winkler, Andreas Beham, and Stefan Wagner

Heuristic and Evolutionary Algorithms Laboratory
School of Informatics, Communications and Media
Upper Austrian University of Applied Sciences, Campus Hagenberg
Softwarepark 11, 4232 Hagenberg, Austria
{maffenze,swinkler,abeham,swagner}@heuristiclab.com

Abstract. This paper discusses some aspects of the general convergence behavior of genetic algorithms. Careful attention is given to how different selection strategies influence the progress of genetic diversity in populations. For being able to observe genetic diversity over time measures are introduced for estimating pairwise similarities as well as similarities among populations; these measures allow different perspectives to the similarity distribution of a genetic algorithm's population during its execution. The similarity distribution of populations is illustrated exemplarily on the basis of some routing problem instances.

1 Introduction

In the theory of genetic algorithms (GAs) population diversity and premature convergence are often considered as closely related topics. The loss of genetic diversity is usually identified as the primary reason for a genetic algorithm to prematurely converge. However, the reduction of genetic diversity is also needed in order to end up with a directed search process towards more promising regions of the search space.

What we would expect from an ideal genetic algorithm is that it loses the alleles of rather poor solutions on the one hand, and on the other hand that it slowly fixes the alleles of highly qualified solutions with respect to a given fitness function. In natural evolution the maintenance of high genetic diversity is important for the ability to adopt to changing environmental conditions. In contrast to this, in artificial evolution, where usually constant optimization goals are to be solved, the reduction of genetic diversity is even necessary for target-oriented search.

In this paper the dynamics of population diversity is documented and discussed in detail on the basis of typical and well known benchmark problem instances of routing problems like the travelling salesman problem (TSP) [3] or the capacitated vehicle routing problem [7] with (CVRPTW) or without time windows (CVRP). Different problem specific similarity measures $similarity(s_1, s_2)$

* The work described in this paper was done within the Josef Ressel centre for heuristic optimization sponsored by the Austrian Research Promotion Agency (FFG).

R. Moreno-Díaz et al. (Eds.): EUROCAST 2009, LNCS 5717, pp. 777–784, 2009.

are introduced for the respective problem representations on the basis of which some test showcases are described. In these tests we examine different parent selection strategies for standard GAs as well as GAs using offspring selection [1].

The rest of the paper is organized as follows: Section 2 discusses some general aspects of selection in evolutionary algorithms and explains the basic principles of offspring selection, and Section 3 introduces the similarity measures used for the comparison of solutions candidates; in Section 4 we document empirical results.

2 Selection Schemes

Concerning guidance of search corresponding to the given fitness function, selection is the driving force of GAs. In contrast to crossover and mutation, selection is completely generic, i.e. independent of the actually employed problem and its representation. A fitness function assigns a score to each individual in a population that indicates the 'quality' of the solution the individual represents. The fitness function is often given as part of the problem description or based upon the objective function.

In the standard genetic algorithm the probability that a chromosome in the current population is selected for reproduction is proportional to its fitness (roulette wheel selection). However, there are also many other ways of accomplishing selection. These include for example linear-rank selection or tournament selection [4], [6]. However, all evenly mentioned GA-selection principles have one thing in common: They all just consider the aspect of sexual selection, i.e. mechanisms of selection only come into play for the selection of parents for reproduction. Offspring selection [1] defies this limitation by considering selection in a more general sense.

2.1 Selection and Selection Pressure

In the population genetics view, especially in the case of not so highly developed species, sexual selection covers only a rather small aspect of selection which appears when individuals have to compete to attract mates for reproduction. The population genetics basic selection model basically considers the selection process in the following way:

random mating → selection → random mating → selection → ......

In other words this means that selection is considered to depend mainly on the probability of surviving of newborn individuals until they reach pubescence which is called viability in the terminology of population genetics. The essential aspect of offspring selection in the interpretation of selection is rarely considered in conventional GA selection.

2.2 Offspring Selection

In principle, offspring selection (OS) acts in the following way: the first selection step chooses the parents for crossover either randomly or in the well-known way of genetic algorithms by proportional, linear-rank, or some kind of tournament selection strategy. After having performed crossover and mutation with the selected parents, offspring selection is inserted: For this purpose, we check the success of the apparently applied reproduction in order to assure the proceeding of genetic search mainly with successful offspring in that way that the used crossover and mutation operators were able to create a child that surpasses its parents' fitness. Therefore, a new parameter, called success ratio ($SuccRatio \in [0,1]$), is introduced. The success ratio gives the quotient of the next population members that have to be generated by successful mating in relation to the total population size. Our adaptation of Rechenberg's success rule, originally stated for the $(1+1)-ES$ [5] for genetic algorithms says that a child is successful if its fitness is better than the fitness of its parents, whereby the meaning of 'better' has to be explained in more detail: is a child better than its parents, if it surpasses the fitness of the weaker, the better, or is it in fact some kind of mean value of both?

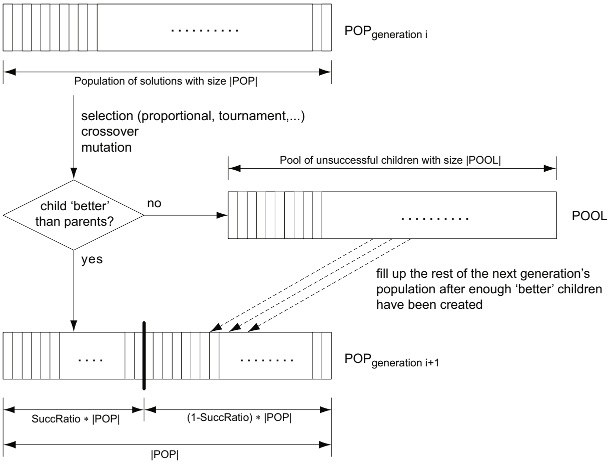

POP$_{generation\ i}$
Population of solutions with size |POP|
selection (proportional, tournament,...)
crossover
mutation
Pool of unsuccessful children with size |POOL|
child 'better' than parents?
no
POOL
yes
fill up the rest of the next generation's population after enough 'better' children have been created
POP$_{generation\ i+1}$
SuccRatio ∗ |POP|
(1-SuccRatio) ∗ |POP|
|POP|

Fig. 1. Flowchart of the embedding of offspring selection into a genetic algorithm

As an answer to this question we claim that an offspring only has to surpass the fitness value of the worst parent in order to be considered as "successful" in the beginning, while as evolution proceeds the child has to be better than a fitness value continuously increasing between the fitness of the weaker and the better parent. As in the case of simulated annealing, this strategy gives a broader search at the beginning, whereas at the end of the search process this operator acts in a more and more directed way. Having filled up the claimed ratio ($SuccRatio$) of the next generation with successful individuals using the success criterion defined above, the rest of the next generation ($(1-SuccRatio)\cdot \mid POP \mid$) is simply filled up with individuals randomly chosen from the pool of individuals that were also created by crossover, but did not reach the success criterion. The actual selection pressure $ActSelPress$ at the end of a single generation is defined by the quotient of individuals that had to be considered until the success ratio was reached and the number of individuals in the population in the following way:

$$ActSelPress = \frac{|POP_{i+1}| + |POOL|}{|POP|}$$

Fig. 1 shows the operating sequence of the above described concepts. With an upper limit of selection pressure $MaxSelPress$ defining the maximum number of children considered for the next generation (as a multiple of the actual population size) that may be produced in order to fulfill the success ratio, this new model also functions as a precise detector of premature convergence:

If it is no longer possible to find a sufficient number of $(SuccRatio \cdot |POP|)$ offspring outperforming their own parents even if $(MaxSelPress \cdot |POP|)$ candidates have been generated, premature convergence has occurred.

As a basic principle of this selection model a higher success ratio causes higher selection pressure. Nevertheless, higher settings of success ratio and therefore of selection pressure do not necessarily cause premature convergence as the preservation of fitter alleles is additionally supported and not only the preservation of fitter individuals. Also it becomes possible within this model to state selection pressure in a very intuitive way that is quite similar to the notation of selection pressure in evolution strategies. Concretely, we define the actual selection pressure as the ratio of individuals that had to be generated in order to fulfill the success ratio to the population size. For example, if we work with a population size of say 100 and it would be necessary to generate 2000 individuals in order to fulfill the success ratio, the actual selection pressure would have a value of 20. Via these means we are in a position to attack several reasons for premature convergence as illustrated in the following sections. Furthermore, this strategy has proven to act as a precise mechanism for self-adaptive selection pressure steering, which is of major importance in the migration phases of parallel evolutionary algorithms.

3 Similarity Measures

The observance of genetic diversity over time is the main objective of this paper. For this reason we apply specific similarity measures in order to monitor and to analyze the diversity and population dynamics. According to the definitions stated in [2] we will use the following problem independent definitions where the concrete definition of $similarity(s_1, s_2)$ has to be stated separately for a certain problem representation:

- **Similarity between two solutions**
  As similarity measures do not have to be symmetric, we use the mean value of the two possible similarity calls and so define a symmetric similarity measure.

$$sim(s_1, s_2) = \frac{similarity(s_1, s_2) + similarity(s_2, s_1)}{2} \tag{1}$$

- **Similarity of a solution** s **within a population** P
  In order to have a measure for the similarity of a certain solution s within a population P at a certain iteration we calculate the average and the maximum similarity of s to all other population members in the following way:

$$meanSim(s,P) = \frac{1}{|P|-1} \sum_{s_* \in P, s_* \neq s} sim(s,s_*) \tag{2}$$

$$maxSim(s,P) = max_{(s_* \in P, s_* \neq s)}(sim(s,s_*)) \tag{3}$$

- **Similarity within a population** P

$$meanSim(P) = \frac{1}{|P|} \sum_{s \in P} meanSim(s,P) \tag{4}$$

$$maxSim(P) = \frac{1}{|P|} \sum_{s \in P} maxSim(s,P) \tag{5}$$

Whereas the similarity definitions stated in the formulae $1 - 5$ do not depend on a concrete problem representation, the similarity itself has to be defined according to the problem representation at hand.

The similarity measure between two TSP-solutions t_1 and t_2 used here is defined as a similarity value *sim* between 0 and 1:

$$sim(t_1,t_2) = \frac{\mid e : e \in E(t_1) \wedge e \in E(t_2) \mid}{\mid E(t_1) \mid} \in [0,1] \tag{6}$$

giving the quotient of the number of common edges in the TSP solutions t_1 and t_2 and the total number of edges. E here denotes the set of edges in a tour. The according distance measure can then be defined as

$$d(t_1,t_2) = 1 - sim(t_1,t_2) \in [0,1] \tag{7}$$

Thus, the similarity or the distance of two concrete TSP solutions can be measured on a linear scale between the values 0 and 1.

The similarity measure for two VRP solutions t_1 and t_2 is calculated in analogy to the TSP similarity using edgewise comparisons. However, as big routes in the VRP are subdivided into smaller routes, a maximum similarity sim_{max} is calculated for each route $r \in t_1$ to all routes $s \in t_2$. These values are summed for all routes r_i and finally divided by the number of routes.

4 Results

The results shown in this section are aimed to just show the basic principle of dynamic diversity analysis for genetic algorithms on the basis of two different GA selection paradigms which are very characteristically. For more sophisticated analyses with tests for more benchmark instances of different combinatorial, real-valued and genetic programming problems performing a sufficient number of test runs for each parameter setting with a sophisticated discussion of the achieved results the interested reader is referred to the book [2].

A very detailed representation of genetic diversity in a population is the statement of pairwise similarities or distances for all members of a population. An appropriate measure, which is provided in the HeuristicLab framework, is to illustrate the similarity as a $n \times n$ matrix where each entry indicates the similarity in form of a grey scaled value. Fig. 2 shows an example: The darker the $(i, j) - th$ entry in the $n \times n$ grid is, the more similar are the two solutions i and j. Not surprisingly, the diagonal entries, which stand for the similarity of solution candidates with themselves, are black indicating maximum similarity.

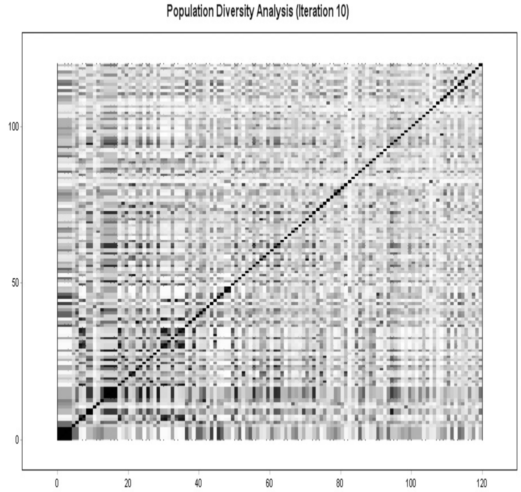


Fig. 2. Degree of similarity/distance for all pairs of solutions in a SGA's population of 120 solution candidates after 10 generations

Unfortunately, this representation is not very well suited for a static monochrome figure. Therefore, the dynamics of this $n \times n$ color grid over the generations is shown in numerous colored animations available at the website of the book [2][1].

For a meaningful figure representation of genetic diversity over time it is necessary to summarize the similarity/distance information of the entire population in a single value. An average value of all n^2 combinations of solution pairs in form of a mean/max similarity value of the entire population as a value between 0 and 1 can be calculated according to the Formulas 2 to 5 stated in section 2. This form of representation allows to display genetic diversity over the generations in a single curve. Small values around 0 indicate low average similarity, i.e., high genetic diversity and vice versa high similarity values of almost 1 indicate little genetic diversity (high similarity) in the population. In the following we

[1] `http://gagp2009.heuristiclab.com`

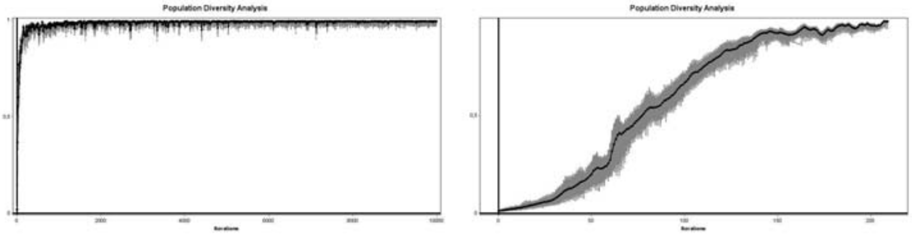
Table 1. Overview of standard GA and offspring selection GA parameters

Parameters for the standard GA		**Parameters for the offspring selection GA**	
Generations	100,000	Population Size	500
Population Size	120	Elitism Rate	1
Elitism Rate	1	Mutation Rate	0.05
Mutation Rate	0.05	Selection Operator	Roulette
Selection Operator	Roulette	Mutation Operator	Simple Inversion
Mutation Operator	Simple Inversion	Success Ratio	0.7
		Maximum Selection Pressure	250

show results of exemplary test runs of GAs applied to the *kroA200* 200 city TSP instance taken from the TSPLib using the parameter settings given in Table 1 and OX crossover.

Fig. 3 shows the genetic diversity curves over the generations for a conventional standard genetic algorithm as well as for a typical offspring selection GA. The gray scaled values of Fig. 3 show the progress of mean similarity values of each individual (compared to all others in the population); average similarity values are represented by solid black lines.

For the standard GA it is observable that the similarity among the solution candidates of a population increases very rapidly causing little genetic diversity already after a couple of generations; it is only mutation which is responsible for reintroducing some new diversity keeping the evolutionary process going. Without mutation the algorithm converges as soon as the genetic diversity of the population is lost, which happens very soon in case of the standard GA. In terms of global solution quality, the finally achieved results with an offspring selection GA are slightly superior to the standard GA and quite close (about $0,5\%$ to 5%) to the global optimum. But for the standard GA this property only holds for well adjusted mutation rates of about 5%. Without mutation the standard GA fails drastically whereas the GA with offspring selection is still able to achieve quite the same solution qualities (about $0,5\%$ to 2% off the global optimum) [2]. The explanation for this behavior is quite simple when we take



,

Fig. 3. Left: Genetic diversity over time in the population of a conventional GA (left figure), **right:** genetic diversity over time in the population of a GA with offspring selection

a look at the genetic diversity over time: in case of offspring selection diversity disappears slowly and controlled whereas in the case of the standard GA diversity is lost very soon and from that time on it is only mutation that keeps evolution running.

Summarizing these results it can be stated for the TSP experiments that the illustration in form of a static figure is certainly some kind of restriction when the dynamics of a system should be observed. For that reason the website of the book [2] contains some additional material showing the dynamics of pairwise similarities for all members of the population (as indicated in Fig. 2) in the form of short motion pictures.

References

1. Affenzeller, M., Wagner, S.: Offspring selection: A new self-adaptive selection scheme for genetic algorithms. In: Ribeiro, B., Albrecht, R.F., Dobnikar, A., Pearson, D.W., Steele, N.C. (eds.) Adaptive and Natural Computing Algorithms. Springer Computer Science, pp. 218–221. Springer, Heidelberg (2005)
2. Affenzeller, M., Winkler, S., Wagner, S., Beham, A.: Genetic Algorithms and Genetic Programming: Modern Concepts and Practical Applications. CRC Press, Boca Raton (2009)
3. Larranaga, P., Kuijpers, C.M.H., Murga, R.H., Inza, I., Dizdarevic, D.: Genetic algorithms for the travelling salesman problem: A review of representations and operators. Artificial Intelligence Review 13, 129–170 (1999)
4. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs. Springer, Heidelberg (1992)
5. Rechenberg, I.: Evolutionsstrategie. Friedrich Frommann Verlag (1973)
6. Schöneburg, E., Heinzmann, F., Feddersen, S.: Genetische Algorithmen und Evolutionsstrategien. Addison-Wesley, Reading (1994)
7. Thangiah, S.R., Potvin, J.-Y., Sun, T.: Heuristic approaches to vehicle routing with backhauls and time windows. International Journal on Computers and Operations Research 23(11), 1043–1057 (1996)